



TECHNICKÁ UNIVERZITA V LIBERCI  
Fakulta mechatroniky, informatiky  
a mezioborových studií ■

# Návrh řídicího systému rehabilitačního rotopedu

## Diplomová práce

*Studijní program:* N2612 – Elektrotechnika a informatika

*Studijní obor:* 3906T001 – Mechatronika

*Autor práce:* **Bc. Tomáš Snětivý**

*Vedoucí práce:* Ing. Josef Černohorský, Ph.D.





TECHNICAL UNIVERSITY OF LIBEREC  
Faculty of Mechatronics, Informatics  
and Interdisciplinary Studies ■

# Control system design of rehabilitation stationary bike

## Diploma thesis

*Study programme:* N2612 – Electrotechnology and informatics

*Study branch:* 3906T001 – Mechatronics

*Author:* **Bc. Tomáš Snětivý**

*Supervisor:* Ing. Josef Černohorský, Ph.D.



## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Tomáš Snětivý**  
Osobní číslo: **M12000260**  
Studijní program: **N2612 Elektrotechnika a informatika**  
Studijní obor: **Mechatronika**  
Název tématu: **Návrh řídicího systému rehabilitačního rotopedu**  
Zadávací katedra: **Ústav mechatroniky a technické informatiky**

### Z á s a d y p r o v y p r a c o v á n í :

1. Proveďte krátkou rešerši dostupných rehabilitačních zařízení založených na kruhovém pohybu.
2. Seznamte se se systémy různých typů brzd využívaných v kondičních rotopedech.
3. Analyzujte elektrodynamickou brzdu rotopedu Kettler.
4. Navrhněte systém řízení rehabilitačního rotopedu.

Rozsah grafických prací: **dle potřeby dokumentace**

Rozsah pracovní zprávy: **40–50 stran**

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:


- [1] Rydlo P.: Řízení elektrických střídavých pohonů, TUL 2007, ISBN 978-80-7273-233-4
- [2] Souček P.: Servomechanismy ve výrobních strojích, ČVUT Praha 2004, ISBN 80-01-02902-6
- [3] Prezentace produktů a technické listy: [www.kalpe.cz](http://www.kalpe.cz), [de.kettler.net](http://de.kettler.net), [www.motomed.com](http://www.motomed.com)
- [4] Technická dokumentace řídicích systémů BR Automation

Vedoucí diplomové práce: **Ing. Josef Černohorský, Ph.D.**  
Ústav mechatroniky a technické informatiky


Konzultant diplomové práce: **prof. Ing. Aleš Richter, CSc.**  
Ústav mechatroniky a technické informatiky

Datum zadání diplomové práce: **10. října 2013**

Termín odevzdání diplomové práce: **16. května 2014**

  
prof. Ing. Václav Kopecký, CSc.  
děkan



  
doc. Ing. Milan Kolář, CSc.  
vedoucí ústavu

V Liberci dne 10. října 2013

## Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.


Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 15. 5. 2014

Podpis: 

## Abstrakt

Cílem této práce byl vývoj rehabilitačního rotopedu, který bude použit při rehabilitaci dolních končetin. Z důvodu urychlení vývoje jsme vyšli z rotopedu Kettler s vířivou brzdou.

Po analýze jsme navrhli zlepšení sensoriky rotopedu a zcela nový řídicí systém na bázi PLC od B&R s dotykovým panelem. Proto jsme na rotoped přidali inkrementální snímač a dva indukční snímače. Díky tomu jsme schopni změřit přesnou rychlost a polohu klik. Dále je možné nastavit libovolnou zátěž do libovolného úseku.

Rotoped má tři hlavní režimy. Režim konstantního výkonu, režim konstantního momentu a eliptický režim řízení momentu. Režim konstantního výkonu v závislosti na rychlosti setrvačnicku nastavuje velikost brzdící síly tak, aby udržel předem zvolený výkon konstantní. Režim konstantního momentu udržuje navolený brzdny moment. Režim eliptické řízení momentu vytváří eliptickou zátěž na pedálech. U této elipsy lze nastavit excentricitu i natočení. Pro ovládání jednotlivých režimů se využívá dotykový panel.

## Klíčová slova

Rehabilitační rotoped  
Režim konstantního výkonu  
Režim řízení momentu  
Elektromagnetická brzda  
PWM  
Vizualizace

## Abstract

The aim of this Diploma thesis was development of rehabilitation stationary bicycle, which could be used for rehabilitation legs as sport training. By reason of increasing development, we came out of Kettler stationary bicycle with eddy current brake.

After analyzing, we proposed improvement in sensors stationary bicycle and use new control system based on PLC of B&R with a touch panel. We add incremental encoder and two inductive sensors. We are able measure velocity and position pedals. We are able to set any load on any segment of revolution.

Stationary bicycle has three main modes. Constant power mode, constant torque mode and elliptic torque mode. Constant power mode is dependent on the speed of the flywheel and set braking force to keep a pre-selected power. Constant torque mode keep the braking torque. Elliptic torque mode generate elliptical load on the pedals. We can change eccentricity of the ellipse and we can set angular offset of the ellipse . To control each modes is used touch panel.

## Keywords

Rehabilitation stationary bicycle  
Constant power mode  
Torque control mode  
electromagnetic brake  
PWM  
Visualization

## Poděkování

Děkuji vedoucímu diplomové práce Ing. Josef Černožorský, Ph.D. za metodické a cíleně orientované vedení při plnění úkolů a za veškerou pomoc při řešení a vypracování diplomové práce. Dále bych rád poděkoval přátelům a rodičům za podporu.



# Obsah

<b>1</b>	<b>Programovatelný logický automat</b>	<b>11</b>
1.1	Historie PLC	11
1.2	Konstrukce PLC	12
1.3	Prvky programovatelného automatu	13
1.3.1	CPU	13
1.3.2	Paměť typu RAM	13
1.3.3	Paměť typu ROM	13
1.3.4	Digitální I/O moduly	13
1.3.5	Analogové I/O moduly	13
1.3.6	Čítačové moduly	14
1.3.7	Komunikační moduly	14
1.4	Činnost PLC	14
1.5	Norma IEC 61 131	14
1.5.1	Typy dat	15
1.5.2	Proměnné	15
1.5.3	Základní stavební bloky programu	16
1.5.4	Funkce	16
1.5.5	Funkční bloky	16
1.5.6	Program	17
1.6	Programovací jazyky PLC	17
1.6.1	Jazyk LD (Ladder Diagram)	17
1.6.2	Jazyk FBD (Function Block Diagram)	17
1.6.3	Jazyk IL (Instruction List)	18
1.6.4	Jazyk ST (Structured Text)	19
<b>2</b>	<b>Rehabilitační robotika</b>	<b>20</b>
2.1	Počátky rehabilitační robotiky	20
2.2	Využití rehabilitační robotiky	20
2.3	Druhy rehabilitačních zařízení založených na kruhovém pohybu	21
2.4	Aktivní cvičení	21
2.5	Pasivní cvičení	22
<b>3</b>	<b>Brzdné systémy rotopedů</b>	<b>23</b>
3.1	Přímé brzdné systémy	23
3.2	Magnetické brzdné systémy	24

3.3	Systémy s odporem vzduchu . . . . .	24
<b>4</b>	<b>Komponenty rehabilitačního rotopedu</b>	<b>26</b>
4.1	Rotoped KETTLER E3 . . . . .	26
4.2	Řídicí systém . . . . .	26
4.2.1	PLC – 4PP065.0351-P74 . . . . .	26
4.2.2	Bus controller – X20BC0083 . . . . .	27
4.2.3	Zdroj 24VDC – X20PS9400 . . . . .	27
4.2.4	I/O modul – X20DI4371 . . . . .	27
4.2.5	Motor modul – X20MM2436 . . . . .	27
4.3	Senzory a čidla . . . . .	28
4.3.1	Indukční snímač – BES M08MI-PSC15B-BV02 . . . . .	28
4.3.2	Inkrementální snímač – LARM IRC-310-5 . . . . .	28
<b>5</b>	<b>Analýza a řízení elektrodynamické brzdy rotopedu KETTLER E3</b>	<b>29</b>
5.1	Analýza elektrodynamické brzdy . . . . .	29
5.2	Řízení elektrodynamické brzdy . . . . .	30
<b>6</b>	<b>Řídicí program</b>	<b>32</b>
6.1	Program – Control . . . . .	32
6.2	Program – Konstantní moment . . . . .	33
6.3	Program – Konstantní výkon . . . . .	33
6.4	Program – Elipsa . . . . .	35
6.5	Program – IOControl . . . . .	36
6.6	Program – HMIControl . . . . .	37
<b>7</b>	<b>Vizualizace</b>	<b>39</b>
7.1	Hlavní strana . . . . .	39
7.2	Setup . . . . .	40
7.3	Vizualizace režimu Konstantní výkon . . . . .	40
7.4	Vizualizace režimu Konstantní moment . . . . .	41
7.5	Vizualizace režimu Elipsa . . . . .	41
	Závěr . . . . .	42
	Literatura . . . . .	43
	Seznam obrázků . . . . .	45
	<b>Příloha</b>	<b>46</b>

## Úvod

Hlavním důvodem, který mě vedl k výběru tohoto tématu diplomové práce, je moje záliba v jízdě na kole a z profesního hlediska je to práce s PLC systémy. K tomuto tématu neexistuje příliš mnoho podobných projektů a vyskytla se zde možnost vytvoření něčeho nového a pro lidi prospěšného.

Tato práce vzniká za spolupráce s Krajskou nemocnicí v Liberci. Cílem bylo vytvoření rehabilitačního rotopedu z běžně dostupného rotopedu. Prvním krokem byl výběr vhodného rotopedu. Ten musel splnit dva hlavní požadavky. Obsahovat elektromagnetickou brzdu a zároveň být cenově dostupný. Tyto požadavky splnil rotoped KETTLER E3. Dalším krokem byl výběr vhodného řídicího systému. Pro tyto potřeby jsem zvolil PLC od firmy B&R. Pro přesný chod a řízení bylo potřeba osadit rotoped i o další čidla a senzory. Po zapojení a oživení celého PLC systému přišla na řadu programová část.

K vytvoření řídicího programu jsem využil software Automation studio, který je přímo od společnosti B&R. Program jsem psal ve strukturovaném textu podle normy IEC 61 131. Programy jsem se snažil členit tak, aby se v každém řešila pouze jedna konkrétní věc a programy tak byly jednoduché a přehledné. Všechny programy jsou koncipovány jako stavové automaty.

Rotoped obsahuje tři hlavní režimy. Režim konstantního výkonu, režim konstantního momentu a režim eliptického řízení momentu. Režim konstantního výkonu v závislosti na rychlosti setrvačníku nastavuje velikost brzdící síly, aby udržoval předem zvolený výkon. Režim konstantního momentu udržuje navolený brzdňý moment. Režim eliptického řízení momentu vytváří eliptickou zátěž na pedálech. U této elipsy lze nastavit excentricitu i natočení. Poslední část se zabývá vizualizací a ovládáním jednotlivých režimů pomocí dotykového panelu.

# 1 Programovatelný logický automat

Nadcházející kapitola popisuje stručně historii **PLC**. Popisuje jeho konstrukci a jednotlivé části. Také je zde ukázán příklad provedení programu a nejzákladnější typy programovacích jazyků.

## 1.1 Historie PLC

Programovatelné automaty, známé také pod zkratkou **PLC** (Programmable Logic Controller), nejsou na našem trhu novinkou. Myšlenka použití počítačů v přímém řízení je jen o něco málo mladší než samy počítače. První pokusy o konstrukci počítačů pro použití v automatizaci se datují již od 50. let minulého století. Tomuto rozvoji pomohlo hlavně rostoucí výkon a spolehlivost počítačů a zároveň jejich klesající cena. To umožnilo na začátku 70. let masivní rozšíření počítačů v oblasti automatizace. Stále se však jednalo o období pokusů a zkoušení. Podle statistik z této doby bylo ovšem v oboru těžkého strojírenství a hutnictví na celém světě zhruba 60 % projektů počítačově automatizovaných systémů neúspěšných. Postupem času se toto číslo stále zmenšovalo díky neustálému pokroku. V cestě širšímu uplatnění počítačů v přímém řízení stála relativně velká cena počítačových systémů. Všechny tyto problémy v jisté míře přetrvávají dodnes, ale již nejsou hlavní překážkou aplikací. Do popředí však neustále, a v současné době čím dál tím více, vystupuje otázka ekonomické efektivity. Hospodárnost ale byla v 70. letech dvacátého století důvodem, který vedl ke konstrukci specializovaných počítačových systémů, jež se i v tehdejších podmínkách již dokázaly efektivně uplatnit v průmyslu.

Specializované počítače našly uplatnění v ovládacích obvodech automatizovaných strojů a zařízení založených na řízení typu ano-ne. Univerzální počítačové systémy byly příliš drahé a pro daný úkol zbytečně složité. To platilo jak pro hardware tak i pro software. V této situaci se na trhu objevily specializované logické procesory a specializované programové vybavení orientované pouze na realizaci ovládacích funkcí. Postupem času však výrazně poklesla cena stále zdokonalovaných univerzálních procesorů, které postupně nahradily jednoúčelové přístroje dosud používané ve speciálních aplikacích. V průběhu 80. let dosáhl vývoj úrovně srovnatelné s dnešním stavem[1].

## 1.2 Konstrukce PLC

Z pohledu konstrukce můžeme PLC rozdělit na dva základní typy a to na *modulární* nebo *kompaktní systém*.

Kompaktní systém (obr. 1.1) je levný a používá se spíše pro jednodušší aplikace. Jeho variabilnost konfigurace je malá nebo žádná. Obvykle má pouze omezený počet digitálních vstupů, výstupů a analogových vstupů či výstupů. Určitou variabilitu umožňuje též použití výměnných násuvných modulů, které se zasouvají do patič na desce plošného spoje[3].



Obrázek 1.1: Kompaktní PLC [2]

Modulární systém (obr. 1.2) je univerzální a používá se pro náročnější aplikace. Hlavní výhodou je možnost přizpůsobení PLC dané úloze. Základem je rám, v jehož levé části je zdroj. V zadní části rámu je vedena interní sběrnice, na níž jsou konektory pro připojení modulu. Jako první se vpravo od modulu zdroje zasouvá modul CPU a pak následují další vstupní/výstupní moduly. K základnímu rámu, ve kterém je CPU, je možno u některých PLC připojit pomocí rozhraní další rozšiřovací rámy. Ty už mají pouze I/O moduly a CPU je společná v základním rámu[3].



Obrázek 1.2: Modulární PLC [2]

## 1.3 Prvky programovatelného automatu

Typické **PLC** z hlediska vnitřního uspořádání obsahuje centrální výpočetní jednotku, systémovou paměť, uživatelskou paměť, binární vstupy/výstupy, analogové vstupy/výstupy a několik dalších modulů jako čítače, komunikační moduly atd.

### 1.3.1 CPU

Centrální procesorová jednotka je jádrem celého programovatelného automatu a určuje jeho výkonnost. Bývá jednaprocesorová, ale může být i víceprocesorová. U víceprocesorových systémů bývá jeden „hlavní“ procesor, který zpracovává program vytvořený uživatelem. Podřízené procesory zajišťují komunikaci po interní sběrnici s jednotkami vstupu/výstupu, komunikaci s dalšími procesory (např. na síti), sběr dat z decentralních periférií a další funkce. V dnešní době není výjimkou ani případ, kdy **CPU** obsahuje **WWW server**[4].

### 1.3.2 Paměť typu RAM

Paměť s přímým nebo libovolným přístupem, to znamená, že ke každé části v paměti je stejný čas přístupu. Jedná se o paměť pro čtení i zápis. Data udržuje pouze podobu napájení. V případě odpojení napájení veškerá data jsou nenávratně ztracena. Využívá se k ukládání řídicího programu a hodnot vstupních, výstupních a pomocných proměnných[4].

### 1.3.3 Paměť typu ROM

Paměť určená pouze pro čtení. Její obsah je dán při výrobě, není závislý na napájení a nelze ji přepsat. Je pomalejší oproti paměti typu **RAM**. Obsahuje systémové programy pro činnost automatu, které uživatel nemůže měnit nebo mazat[4].

### 1.3.4 Digitální I/O moduly

Digitální vstupní moduly slouží k připojování prvků, které mají pouze dva stavy a to zapnuto a vypnuto. často se jedná např. o tlačítka, přepínače, senzory doteku nebo přiblížení, dvouhodnotové senzory tlaku, hladiny, teploty apod. K digitálním výstupním modulům se obvykle připojují nejrozumnější akční členy s dvouhodnotovým charakterem vstupního signálu. Mohou to být např. různá optická i akustická signalizační zařízení nebo cívky relé, stykačů, solenoidových ventilů, elektromagneticky pneumatických či hydraulických ovládaných rozvaděčů[4].

### 1.3.5 Analogové I/O moduly

Analogové vstupní a výstupní moduly se používají v případech, kdy nám nestačí stavy (zapnuto/vypnuto). Důležitou součástí analogových modulů je **A/D** převodník, který převádí analogový signál (napětí, proud) na digitální signál (číselné hodnoty).

Přesnost převodu závisí na bitové hloubce A/D převodníku. Z pravidla platí, čím vyšší počet bitů tím je A/D převodník pomalejší a naopak. K analogovým vstupům lze připojit například snímače teploty, vlhkosti, tlaku, síly, hladiny, rychlosti, ale i většinu inteligentních přístrojů s analogovými výstupy. Analogové výstupní jednotky slouží pro ovládání různých akčních členů se spojitým charakterem vstupního signálu, jako např. spojitě servopohony, frekvenční měniče, ale třeba i ručkové měřicí přístroje[4].

### 1.3.6 Čítačové moduly

Čítačové moduly jsou určeny k čítání pulsů, jejichž perioda je srovnatelná nebo kratší, než je smyčka programu v PLC. Bývají k dispozici v provedení pro připojení univerzálních signálů, inkrementálních snímačů nebo absolutních snímačů. Všechny programovatelné automaty jsou také vybaveny softwarovými čítači, které se s výhodou používají v situacích, kdy není zapotřebí použití čítačových modulů[4].

### 1.3.7 Komunikační moduly

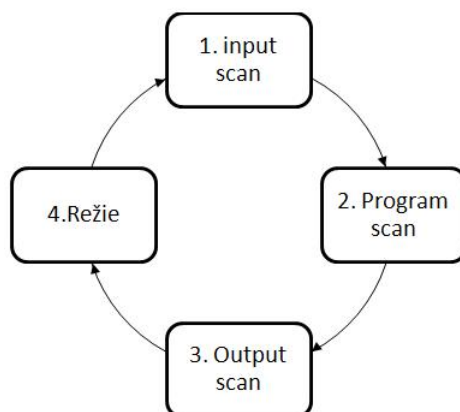
Hlavní výhodou a důvodem, proč se PLC systémy tak rozšířily v průmyslu, je právě schopnost decentralizace. Komunikaci mezi těmito vzdálenými moduly zajišťují právě komunikační moduly. Díky tomu může PLC komunikovat se vzdálenými vstupy a výstupy, s podsystémy, se souřadnými i nadřizenými systémy, s operátorským panelem, s počítači a jejich sítěmi a tak vytvářet distribuované systémy. Nejrozšířenějšími komunikačními rozhraními jsou Ethernet, RS 232 a Powerlink[4].

## 1.4 Činnost PLC

PLC pracuje cyklicky, to znamená, že od spuštění neustále vykonává nahraný program až do vypnutí. Pro ukázkou je zde obrázek (obr. 1.3), který znázorňuje právě tento cyklus. Na začátku každého cyklu si PLC uloží do paměti stav všech vstupů. Na základě stavů vstupu vykoná řídicí program. Po ukončení programu aktualizuje stav výstupů. Nakonec se provádí část režie, kde se provádí komunikace a servisní služby jako například aktualizace systémových registrů a proměnných časovačů, nulování watchdog timeru atd[3].

## 1.5 Norma IEC 61 131

Norma IEC 61 131 pro programovatelné řídicí systémy má sedm základních částí a představuje souhrn požadavků na moderní řídicí systémy. Je nezávislá na konkrétní organizaci či firmě a má širokou mezinárodní podporu. Jednotlivé části normy jsou věnovány jak technickému, tak programovému vybavení těchto systémů. V ČR byly přijaty jednotlivé části této normy pod následujícími čísly a názvy:



Obrázek 1.3: Činnost PLC

- ČSN EN 61 131-1 Programovatelné řídicí jednotky-Část 1: Všeobecné informace
- ČSN EN 61 131-2 Programovatelné řídicí jednotky-Část 2: Požadavky na zařízení a zkoušky
- ČSN EN 61 131-3 Programovatelné řídicí jednotky-Část 3: Programovací jazyky
- ČSN EN 61 131-4 Programovatelné řídicí jednotky-Část 4: Podpora uživatelů
- ČSN EN 61 131-5 Programovatelné řídicí jednotky-Část 5: Komunikace
- ČSN EN 61 131-7 Programovatelné řídicí jednotky-Část 7: Programování fuzzy řízení

V Evropské unii jsou tyto normy přijaty pod číslem EN IEC 61 131[5].

### 1.5.1 Typy dat

V rámci zachování kompatibility jsou definovány typy dat. Definování datových typů napomáhá prevenci chyb v samém počátku tvorby projektu. Je nutné definovat typy všech použitých parametrů. Běžné datové typy jsou **BOOL**, **BYTE**, **WORD**, **INT**, **REAL**, **DATE**, **TIME**, **STRING** atd. Z těchto základních datových typů je pak možné odvozovat vlastní uživatelské datové typy, tzv. odvozené datové typy. Tímto způsobem můžeme např. definovat jako samostatný datový typ analogový vstupní kanál a opakovaně ho používat pod definovaným jménem.[5]

### 1.5.2 Proměnné

Proměnné mohou být přiřazeny explicitně k hardwarovým adresám (např. vstupům, výstupům) pouze v konfiguracích, zdrojích nebo programech. Tímto způsobem je dosaženo vysokého stupně hardwarové nezávislosti a možnosti opakovaného využití softwaru na různých hardwarových platformách. Oblast působnosti proměnných



je běžně omezena pouze na tu programovou organizační jednotku, ve které byly deklarovány (lokální proměnné). To znamená, že jejich jména mohou být používána v jiných částech bez omezení. Tímto opatřením dojde k eliminaci řady dalších chyb. Pokud mají mít proměnné globální působnost, např. v rámci celého projektu, pak musí být jako globální deklarovány. Aby bylo možné správně nastavit počáteční stav procesu nebo stroje, může být parametrům přiřazena počáteční hodnota při startu nebo studeném restartu[5].

### 1.5.3 Základní stavební bloky programu

Základním pojmem při programování podle normy IEC 61 131-3 je termín Programová Organizační Jednotka nebo zkráceně **POU** (Program Organisation Unit). Jak vyplývá z názvu, **POU** je nejmenší nezávislá část uživatelského programu. **POU** mohou být dodávány od výrobce řídicího systému nebo je může napsat uživatel. Každá **POU** může volat další **POU** a při tomto volání může volitelně předávat volané **POU** jeden nebo více parametrů. Existují tři základní typy **POU**: funkce, funkční blok, program[5].

### 1.5.4 Funkce

IEC 61 131-3 definuje standardní funkce a uživatelem definované funkce. Standardní funkce jsou např. **ADD** pro sčítání, **ABS** pro absolutní hodnotu, **SQRT** pro odmocninu, **SIN** pro sinus a **COS** pro cosinus. Jakmile jsou jednou definovány nové uživatelské funkce, mohou být používány opakovaně[5].

### 1.5.5 Funkční bloky

Na funkční bloky se můžeme dívat jako na integrované obvody, které reprezentují hardwarové řešení specializované řídicí funkce. Obsahují algoritmy i data, takže mohou zachovávat informaci o minulosti (tím se liší od funkcí). Mají jasně definované rozhraní a skryté vnitřní proměnné, podobně jako integrovaný obvod nebo černá skříňka. Umožňují tím jednoznačně oddělit různé úrovně programátorů nebo obslužného personálu. Klasickými příklady funkčního bloku jsou např. regulační smyčka pro teplotu nebo **PID** regulátor.

Jakmile je jednou funkční blok definován, může být používán opakovaně v daném programu, nebo v jiném programu, nebo dokonce i v jiném projektu. Je tedy univerzální a mnohonásobně použitelný. Funkční bloky mohou být zapsány v libovolném z jazyků definovaném v normě. Mohou být tedy plně definovány uživatelem. Odvozené funkční bloky jsou založeny na standardních funkčních blocích, ale v rámci pravidel normy je možno vytvářet i zcela nové zákaznické funkční bloky. Interface funkcí a funkčních bloků je popsán stejným způsobem. Mezi deklarací označující název bloku a deklarací pro konec bloku je uveden soupis deklarací vstupních proměnných, výstupních proměnných a vlastní kód v tzv. těle bloku[5].

### 1.5.6 Program

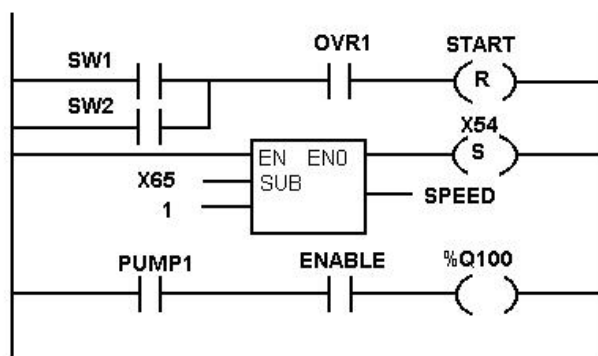
Posledním typem **POU** je program, který představuje vrcholovou programovou jednotku v uživatelském programu. Centrální jednotka **PLC** může zpracovávat více programů a programovací jazyk **ST** obsahuje prostředky pro definice spouštění programů (v jaké periodě vykonávat program, s jakou prioritou, apod.)[5].

## 1.6 Programovací jazyky PLC

Norma IEC 61131-3 definuje 4 typy programovacích jazyků. Dva jsou grafické (**LD**, **FBD**) a dva textové (**ST**, **IL**). Někdy se uvádí i pátý programovací jazyk a to **SFC** (Sequential Function Chart), který však není v normě zařazen mezi jazyky, ale mezi tzv. společnými prvky, neboť tvoří jakousi nadstavbu pro strukturování celé aplikace. Některé vývojové systémy umožňují programování ještě v dalších jazycích, např. v jazyce **C**. [6]

### 1.6.1 Jazyk LD (Ladder Diagram)

Grafický jazyk **LD** (obr. 1.4) je někdy také nazýván jazykem kontaktních schémat a je založen na grafické reprezentaci reléové logiky. Organizační jednotka programu je vyjádřena sítí propojených grafických prvků. Sít v jazyku **LD** je zleva i zprava ohraničena svislými čarami, které se nazývají levá a pravá napájecí sběrnice. Mezi nimi je tzv. příčka, která může být rozvětvena. Každý úsek příčky, vodorovný nebo svislý, může být ve stavu *on* nebo *off*. Do příček mohou být zapojeny kontakty v sérioparalelních kombinacích. Mezi kontakty mohou být zapojeny spínače, rozpínače, cívky, funkce a funkční bloky [6].

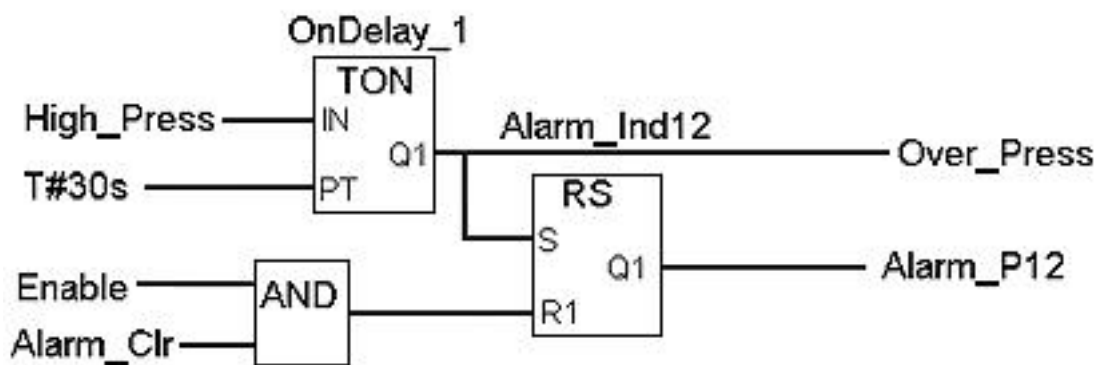


Obrázek 1.4: Ukázka příčkového diagramu [7]

### 1.6.2 Jazyk FBD (Function Block Diagram)

Druhým grafickým jazykem je **FBD** (obr. 1.5), který vyjadřuje chování funkcí, funkčních bloků a programů jako soubor vzájemně provázaných grafických bloků podobně

jako v elektronických obvodových diagramech. Jde o systém prvků, které zpracovávají signály. Často se zde používají standardní funkční bloky, jako jsou např. bistabilní prvky (paměti s dominantním vypnutím nebo sepnutím), prvky pro detekci náběžné a sestupné hrany, čítače, časovače a komunikační bloky definované v normě IEC 1131-5. Podle potřeby jsou doplňovány speciální bloky a každá firma nabízí ve svém programovacím prostředí poněkud odlišný soubor bloků (např. spínací hodiny týdenní, roční, generátory impulzů, komparátory apod.)[6].



Obrázek 1.5: Ukázka funkčního blokového schématu [8]

### 1.6.3 Jazyk IL (Instruction List)

IL (obr. 1.6) patří do skupiny textových jazyků. Bývá označován také jako jazyk pokynů, seznam instrukcí a je velice podobný assembleru. Programová organizační jednotka je složena ze sekvence instrukcí, z nichž každá začíná na novém řádku, může obsahovat také komentář. Pomocí modifikátorů se vyjadřují negace, podmíněnost a nepodmíněnost instrukce skoků, volání a návratů[6].

0	BLK	%C8
1	LDF	%I0.1
2	R	
3	LD	%I0.2
4	AND	%M0
5	CU	
6	OUT_BLK	
7	LD	D
8	AND	%M1
9	ST	%Q0.4
10	END_BLK	

Obrázek 1.6: Ukázka seznamu instrukcí [12]

### 1.6.4 Jazyk ST (Structured Text)

Textový jazyk **ST** (obr. 1.7) je výkonný vyšší programovací jazyk, který má kořeny v jazycích **Pascal** a **C**. Je to velmi účinný nástroj pro zápis náročných algoritmů a pro vytváření komplexních funkčních bloků, které pak mohou být použity v libovolném programovacím jazyku. Syntaxe jazyka je dána povolenými výrazy a příkazy. Vyhodnocením výrazu vyjde hodnota v některém z definovaných datových typů. Výraz se skládá z operátorů a operandů. Operandem může být konstanta, proměnná, funkce nebo jiný výraz. Operátory pro jazyk **ST** jsou definovány pro sedmnáct typů operací (vyhodnocení funkce, negace, násobení, booleovské funkce **AND**, **XOR** a **OR** apod.). Je definováno deset typů příkazů (přiřazení, vyvolání funkce, návrat, výběr apod.). Příkazy jsou odděleny středníkem a může jich být více na jednom řádku.[6].

```
IF (yVal < 0) THEN
  yVal := yVal + offset;
  bottom := yVal + offset;
ELSE
  IF (xVal < 470) THEN
    xVal := xVal + offset;
  END_IF
  IF (bottom > -250) THEN
    bottom := bottom - offset;
  END_IF
END_IF
ST: Structured Text
```

Obrázek 1.7: Ukázka strukturovaného textu [9]

## 2 Rehabilitační robotika

### 2.1 Počátky rehabilitační robotiky

Počátky sahají do doby Americké občanské války, kdy vlivem velkého počtu amputací extrémně vzrostl zájem o protézy. Amerika tak byla donucena vynaložit nemalé finanční prostředky pro vývoj protéz. Po první světové válce v roce 1917 došlo k založení Americké ortopedické a protetické asociace. Na konci druhé světové války čelila americká vláda nespokojenosti z nedostatku technologií. To zapříčinilo, že byly uzavřeny smlouvy s vojenskými společnostmi, které otevřely cestu k vývoji moderních protéz a dalších technologií[10].

Počátky rehabilitačního inženýrství sahají do 60. let, kdy začal vývoj poháněného robotického exoskeletu. Tímto výzkumem se zabývaly převážně Spojené státy a Jugoslávie. Hlavní důraz byl kladen na zlepšení rozhraní stroje a člověka. Exoskelet je robotický pomocný mechanismus s klouby a linkami odpovídající lidskému tělu. Člověk pak slouží jako poskytovatel řídicích signálů a exoskelet poskytuje většinu ze síly potřebné k plnění úkolu. V roce 1965 americká vláda představila vynález Hardiman. Byl to exoskelet umožňující nositeli uzvednout až 680 kg[10].

Souběžně s vývojem exoskeletu probíhal vývoj robotických manipulátorů. První sestrojený rehabilitační manipulátor byl v roce 1960 tzv. **CASE** manipulátor. Byla to napájená ortéza se čtyřmi stupni volnosti, která pomáhala pohybovat s pacientovou ochrnutou paží. Většina tehdy vytvořených rehabilitačních manipulátorů nebyla primárně určena pro rehabilitaci, ale byly to předělané průmyslové manipulátory. Všechny tyto manipulátory byly nemobilní. A právě tento fakt se vědci snažili změnit. Nejúspěšnějším mobilním manipulátorem se stal Manipulátor Manus. Jednalo se o manipulátor namontovaný na elektrický vozík. Prodávat se začal v roce 1990[11].

### 2.2 Využití rehabilitační robotiky

Uplatnění robotické technologie nalézají ve všech lékařských oborech. Nejrozšířenějším typem je rehabilitační robotika, která zahrnuje pomocné roboty, ortézy, protézy a řadu dalších. Cílem tohoto oboru je vývoj technologií zvyšujících výkonnost lidského těla či usnadňující aktivity každodenního života pacientů. Přestože rehabilitační robotika zasahuje do více typů terapií, největšího pokroku v poslední době dosahuje v oblasti neurologické, která se snaží obnovit nebo zlepšit svalovou kontrolu u pacientů[10].

Roboti by měli být plně nastavitelní, aby umožnili různé pohyby, popřípadě se přizpůsobili fyzickým schopnostem pacienta. Dále by měli být schopni detekovat úmysl a podpořit pacienta v pohybu. V poslední době se začíná využívat virtuální reality, která pomáhá jak s motivací tak se zlepšením orientace v prostoru, koordinace pohybu a nácviku chůze[10].

## 2.3 Druhy rehabilitačních zařízení založených na kruhovém pohybu

Při hledání na internetu lze zjistit, že problematikou rehabilitačních rotopedů se zabývá pouze malý počet firem. Je to nejspíše způsobené tím, že se nejedná o přístroje pro širokou veřejnost. Ceny rehabilitačních rotopedů jsou až třikrát vyšší než klasické rotopedy. Často obsahují složité řídicí programy, které umí např. rozeznat křeče a následně na tyto podněty reagovat.

Obecně platí, že rehabilitační rotopedy nejsou uzpůsobeny pro cvičení s velkými zátěžemi, ale specializují se na malé a plynulé změny zátěže.

Můžeme je v zásadě rozdělit podle typu cvičení, a to na pasivní a aktivní cvičení nebo podle zaměření na určitý typ končetin.

Pro rehabilitaci dolních končetin se používají klasické rotopedy nebo tzv. přenosné pedály viz (obr. 2.1), které je možné položit na stůl nebo na podlahu a procvičovat tak dolní nebo horní končetiny. Posledním typem jsou rehabilitační rotopedy uzpůsobeny k současnému pohybu jak dolních, tak i horních končetin. Existují i modifikace těchto rotopedů, které jsou přizpůsobeny lidem na vozíčku nebo procvičování přímo v posteli[13].



Obrázek 2.1: Příklad přenosného rehabilitačního zařízení [13]

## 2.4 Aktivní cvičení

Při aktivním cvičení pacient využívá pouze své síly. To znamená, že pacientovi při cvičení nic nepomáhá. Pro tyto účely se využívají rotopedy bez elektromotoru a jedná se v podstatě o klasické rotopedy, které mají ovšem daleko propracovanější programy pro cvičení. Na trhu se dají pořídit okolo 10000 Kč.

## 2.5 Pasivní cvičení

Pasivní cvičení nastává, když pacient není schopen cvičit sám nebo je schopen jen částečně s vnější pomocí. Existují dva typy rehabilitačních rotopedů pro pasivní cvičení. První typ je poháněn elektromotorem. Elektromotor pomáhá pacientovi hlavně při rozjezdu. Jsou-li aktivní síly cvičícího slabé, přístroj se automaticky přepne do režimu asistenční pomoci a pomáhá cvičícímu rozvíjet jeho vlastní síly. Nedokáže-li cvičící vyvinout žádnou vlastní sílu, přístroj se automaticky přepne do režimu pasivního cvičení a pohybuje končetinami místo cvičícího. Obousměrné přepínání režimů cvičení se děje automaticky na základě nepřetržitého vyhodnocování průběhu cvičení. Dokáže také detekovat křeče a přepnout tak na speciální program na uvolňování křečí. Cena těchto rehabilitačních přístrojů se pohybuje kolem 60000 Kč[13].

Dalším příkladem jsou přístroje poháněné silou paží cvičícího nebo asistenta sedícího při cvičení proti cvičícímu viz (obr. 2.2). Díky optimálnímu převodu lze cvičit bez vynaložení velké síly paží a procvičování dolních končetin tudíž zvládne téměř každý. Velkou výhodou těchto zařízení je podstatně nižší cena ve srovnání s přístrojem poháněným motorem. Cena se pohybuje okolo 20000 Kč. Přístroje jsou vybaveny setrvačníkem umožňujícím rovnoměrný chod přístroje a brzdou, která umožňuje regulaci brzdné síly, což dovoluje i samostatný aktivní pohyb[13].



Obrázek 2.2: Příklad rehabilitačního zařízení poháněný silou asistenta [13]

## 3 Brzdné systémy rotopedů

Většina rotopedů je postavena na základě setrvačníku. Setrvačnick je v podstatě velký těžký železný disk, který je spojen pomocí řemene k pedálům. Při šlapání se tak roztáčí a vytváří brzdny moment. Největší odpor setrvačnicku je cítit při rozjezdu. Ovšem samotná brzdná síla setrvačnicku nestačí. Proto se využívá dalšího zařízení pro vytvoření většího brzdného momentu.

Brzdné systémy lze rozdělit do tří skupin, a to na přímé brzdné systémy, magnetické systémy a systémy s odporem vzduchu.

### 3.1 Přímé brzdné systémy

Brzdnou sílu je vytvářena pomocí třecí síly působící na setrvačnick. V zásadě existují dva druhy přímých brzdných systémů. Páskové, které obepínají setrvačnick a pomocí napínání pásky zvyšují brzdny moment. V dnešní době se již nepoužívají.

Čelistové brzdné systémy, které mají kotoučovou brzdu s brzdnymi destičkami. Ty tlačí na setrvačnick a tím jej brzdí. Vlivem tření dochází k opotřebování destiček. Z tohoto důvodu je nutné čas od času destičky vyměnit. Velkou předností rotopedů s brzdnymi destičkami je jejich schopnost nastavit velmi vysoké zátěže. Nevýhoda těchto systémů je v jejich vyšší hlučnosti[16].

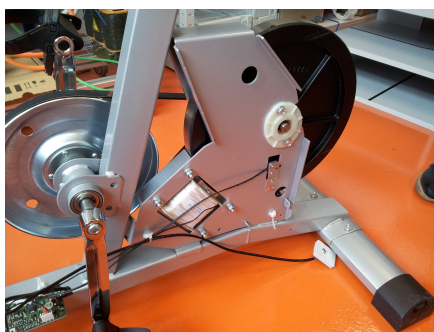


Obrázek 3.1: Příklad čelistové brzdy [16]



## 3.2 Magnetické brzdné systémy

V současnosti se začínají hojně využívat magnetické brzdné systémy. Jejich hlavní výhodou je tichý chod. V levnějších rotopedech se využívá permanentní magnet, který přiblížováním a oddalováním od setrvačnicku mění brzdnou sílu. Rotopedy s magnetickým brzdným systémem neumožňují nastavit tak vysoké zátěže jako rotopedy s čelistovým brzdným systémem. Pro běžné domácí cvičení je ovšem magnetický brzdný systém zcela postačující. U dražších rotopedů se vyskytují indukční brzdné systémy. Zde je odpor vytvářen velikostí elektrického proudu, který prochází cívkou, a ta se pak chová jako magnet. Rotopedy s indukčním brzdným systémem dovolují velmi přesné nastavení zátěže. Výhodou těchto systémů je jejich dlouhá životnost[16].



Obrázek 3.2: Příklad elektromagnetické brzdy

## 3.3 Systémy s odporem vzduchu

Rotopedy založené na systému s odporem vzduchu (obr. 3.3) se liší od normálních rotopedů. Tyto rotopedy nemají setrvačnicku, ale mají ventilátor. Ventilátor je připojen k pedálům pomocí řemenu. To znamená, že čím rychleji se otáčí pedály, tím rychleji se otáčí ventilátor. Velikost odporové síly je závislá na rychlosti pedálů. Z tohoto důvodu tyto rotopedy nepotřebují žádné řízení pro měnění brzdné síly. Pro zmenšení odporu stačí snížit frekvenci šlapání[15].



Obrázek 3.3: Příklad brzdy pomocí odporu vzduchu [14]

## 4 Komponenty rehabilitačního rotopedu

### 4.1 Rotoped KETTLER E3

Hlavním kritériem při výběru rotopedu bylo, aby obsahoval elektroindukční brzdový systém. Důvod výběru tohoto typu rotopedů byl jednoduchý. Tyto brzdové systémy jsou přesné a dají se jednoduše řídit pomocí proudu. Další parametry, které rozhodovaly při výběru, byla cena a dostupnost rotopedu. Všechny tyto parametry splnil právě Rotoped **KETTLER E3**. Níže jsou uvedeny další parametry rotopedu **KETTLER**.

- Elektroindukční systém s tichým chodem
- Setrvačnick o hmotnosti 6 kg
- Střed osazen průmyslovými ložisky **SKF**
- Elektronická regulace výkonu nezávislá na otáčkách 25 - 400 W (po 5 W)
- Průmyslový, podélně drážkovaný plochý řemen pro tichý chod, převod 1:8
- Komfortní gelové sedlo s horizontálním a vertikálním posuvem
- Přijímač signálu **POLAR**
- Stabilní rám ze značkové oceli o nosnosti 150 kg
- Vzdálenost pedálů od sedla 72 až 100cm
- Ergonomická řídítka s nastavitelným sklonem v rozmezí 360 stupňů

### 4.2 Řídicí systém

Tato část se zabývá popisem řídicího systému rehabilitačního rotopedu. Je zde vysvětleno, k čemu se daný modul využívá a jaké signály zpracovává, popřípadě vysílá.

#### 4.2.1 PLC – 4PP065.0351-P74

Jedná se o tzv. kombo, kdy **PLC** obsahuje i dotykový **TFT** display. Dotykový displej má 3.5" úhlopříčku s rozlišením 320 × 240 px. **PLC** má procesor s taktem 500 MHz a 128 MB **RAM** pamětí. Pro komunikaci je možné využít **USB** nebo **Ethernet**[17].



Obrázek 4.1: Ukázka řídicího PLC [17]

#### 4.2.2 Bus controller – X20BC0083

Tento modul propojuje Power Panel s dalšími přídatnými moduly pomocí rozhraní **POWERLINK**. **POWERLINK** je sběrnice založená na bázi Ethernetu. Původně vyvinutá společností **B&R**. Nyní ji dále rozvíjí nezávislá společnost Ethernet Powerlink Standardization Group (EPSEG). Specifikace protokolu je otevřená a volně přístupná[17].

#### 4.2.3 Zdroj 24VDC – X20PS9400

Zdroj stejnosměrného napětí 24V se využívá k napájení sběrnice, přídatných modulů a Power Panelu[17].

#### 4.2.4 I/O modul – X20DI4371

Modul obsahující digitální vstupy slouží k připojení dvou indukčních senzorů[17].

- 4 digitální vstupy
- 3 drátové zapojení
- 4 čítačové vstupy s frekvencí čítání až 1 kHz
- typ zapojení Sink

#### 4.2.5 Motor modul – X20MM2436

U motor modulu využíváme jeden výstup v režimu pro řízení indukčních zátěží, kterým řídíme pomocí **PWM** elektromagnetickou brzdu. Dále využíváme jeden vstup, kde je zapojen inkrementální snímač[17].

- 2×výstup (H můstek) s **PWM** řízením od 24 do 39 VDC
- Nominální proud 3 A

- Frekvence od 15 Hz do 50 kHz, 16-bit
- Rozlišení **PWM**, 15-bit, minimum 10 ns
- 2×vstup 24 V pro inkrementální snímač
- typ zapojení Sink

## 4.3 Senzory a čidla

Pro naše potřeby řízení jsme museli osadit rehabilitační rotoped přídatnými snímači, protože stávající snímače buď chyběly, nebo měly nedostatečné rozlišení.

### 4.3.1 Indukční snímač – BES M08MI-PSC15B-BV02

Na rehabilitačním rotopedu využíváme dva tyto snímače. Jeden je umístěn u setrvačníku, pomocí kterého měříme jeho rychlost. Druhý je umístěn u klik, kde ho využíváme jako referenční polohu a také jako korekci při měření rychlosti a polohy klik[18].

- Napájení 24 VDC
- Zapojení 3 drátové
- Dosah 1,5 mm
- Pracovní frekvence 1000 Hz
- Opakovatelnost < 5%

### 4.3.2 Inkrementální snímač – LARM IRC-310-5

Tento snímač je nainstalován na klikách a s jeho pomocí můžeme měřit přesnou polohu a rychlost klik. To je důležité hlavně pro režim Elipsa, kde potřebujeme znát přesnou polohu klik pro výpočet zátěže[19].

- Maximální otáčky 10000 ot/min
- Maximální úhlové zrychlení  $40000 \text{ rad} \cdot \text{s}^{-2}$
- 1500 pulzů na otáčku

## 5 Analýza a řízení elektrodynamické brzdy rotopedu KETTLER E3

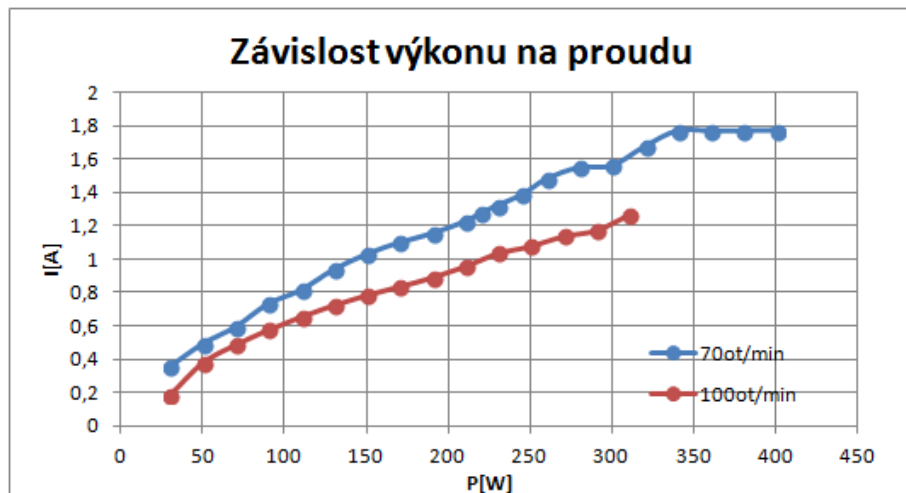
Tato kapitola se zabývá popisem a chování elektrodynamické brzdy v závislosti protékaného proudu na výstupním momentu. Další část této kapitoly se zaměřuje na popis řízení elektrodynamické brzdy.

### 5.1 Analýza elektrodynamické brzdy

Tento úkol se ze začátku jevil jako velice jednoduchý, ale při měření jsme narazili na několik problémů. Největší problém byl ten, že měření bylo zatíženo velkými nepřesnostmi. Ty vznikaly hlavně díky nedostatečnému rozlišení při měření rychlosti od senzoru z původního rotopedu, který měl pouze jeden pulz na otáčku. Z tohoto důvodu řídicí jednotka nedokázala správně vypočítat rychlost otáčení setrvačníku a řídicí systém rotopedu neustále musel měnit proud do elektrodynamické brzdy. Tento problém jsme vyřešili tak, že jsme měřili vždy průměrnou hodnotu proudu pomocí osciloskopu a měření několikrát opakovali. Právě opakovatelnost byla dalším problémem, protože bylo potřeba pokaždé proměřit hodnoty proudu při zátěži od 0 W do 400 W po kroku 10 W, což bylo fyzicky hodně náročné a my jsme potřebovali navíc naměřit hodnoty pro několik rychlostí.

Níže jsou zobrazeny dva grafy. Na prvním (obr. 5.1) je zachycena závislost výkonu brzdě síly na protékajícím proudu brzdou. V druhém grafu (obr. 5.2) je vyobrazena závislost brzděho momentu na protékajícím proudu brzdou. Z grafů je vidět, že jednotlivé charakteristiky nejsou přesně lineární, zvláště při malých zátěžích. Všechny tyto nepřesnosti jsou způsobeny jednak nepřesným měřením rychlosti otáčení setrvačníku viz výše, a dále také nepravidelným šlapáním. Naměřené charakteristiky jsme proložili přímkami a z výsledných rovnic jsme vytvořili finální rovnici pro přepočet mezi momentem a proudem, viz rovnice (1) .

$$y = 0,0287 \cdot x + 0,3564 \quad (1)$$

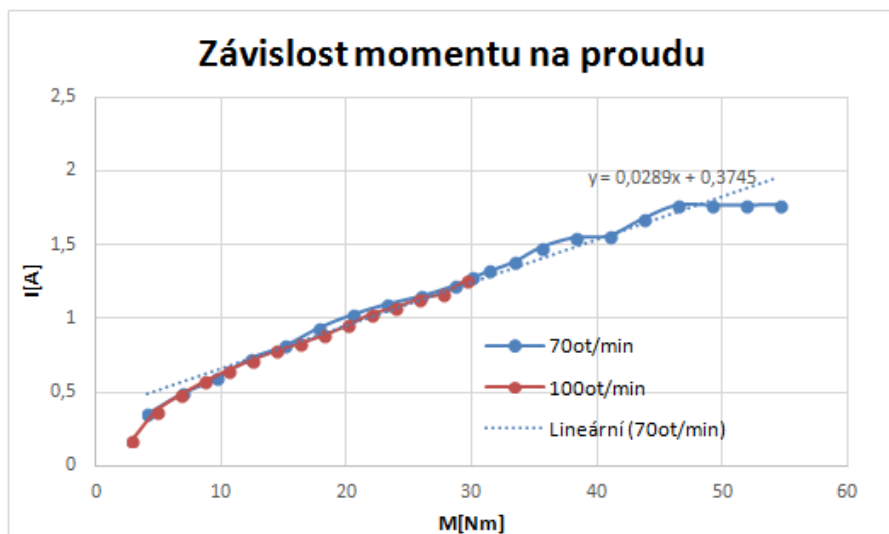


Obrázek 5.1: Ukázka závislosti výkonu na proudu

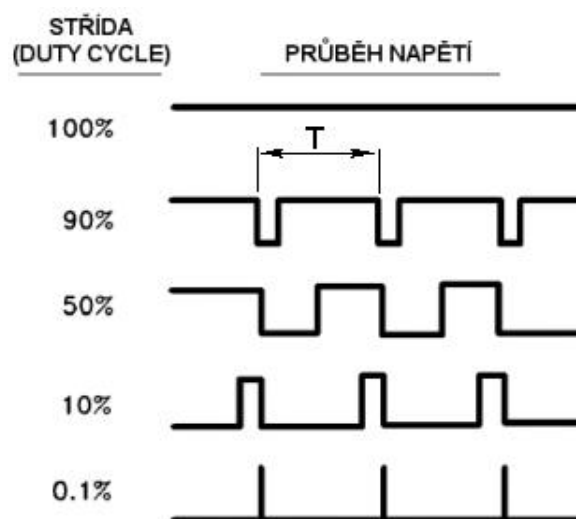
## 5.2 Řízení elektrodynamické brzdy

K řízení velikosti proudu brzdou jsme se rozhodli použít **PWM** modulaci. Jedná se o modulaci, kdy u signálu s konstantní periodou  $T$  se mění střída proudu (tj. poměr délky sepnutého výstupu ku délce vypnutého výstupu v jedné periodě). Střída se uvádí někdy jako poměr (1:1, 2:1, 1:5 atd.), kdy je nutné uvést které číslo představuje impuls a které mezeru. Někdy se střída vyjadřuje procentuálně (100%, 50%, 0,1% atd.), kde 100% představuje ideální poměr 1:0, 50% poměr 1:1 atd. Poměr délky impulsu ku délce mezery bývá v zahraniční literatuře nazýván **Duty Cycle**. Pro názornost princip **PWM** modulace je znázorněn na (obr. 5.3)

Původně jsme chtěli k řízení použít obyčejný **DO** modul a k němu vytvořit krátký program, aby pracoval v režimu **PWM** modulace. Bohužel při zkoušení tohoto typu řízení jsme zjistili, že frekvence vykonávání jednotlivých cyklických programů nám nedovoluje dostatečné a přesné řízení. Mohli jsme nastavit buď malou periodu, v tom případě byl však problém s rozlišením jednotlivých kroků, nebo nastavit větší periodu, ale při takovém postupu už brzda nefungovala správně. Proto jsme pro řízení zvolili motor modul (**X20MM2436**), který má přímo zabudovaný výstup pro řízení indukčních zátěží pomocí **PWM**.



Obrázek 5.2: Ukázka závislosti momentu na proudu



Obrázek 5.3: Princip PWM modulace



## 6 Řídicí program

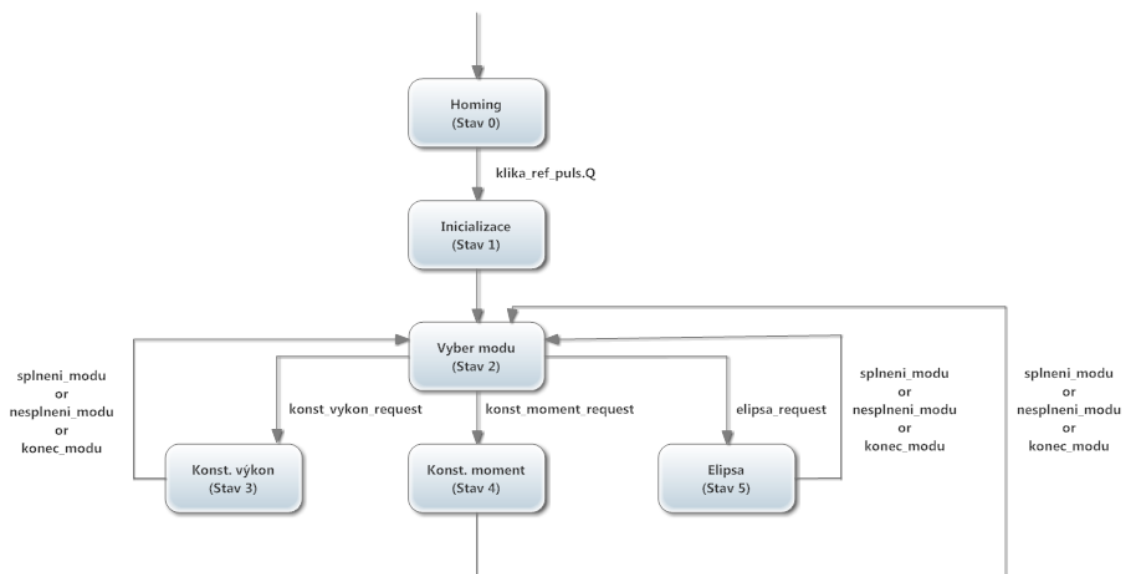
Řídicí program je napsán ve vývojovém prostředí Automation studio, který je přímo od společnosti **B&R**. Program jsem psal ve strukturovaném textu podle normy IEC 61 131. Pro co největší přehlednost jsem se snažil rozdělit jednotlivé problémy na menší části a v každém programu řešit pouze konkrétní problém. Vzhledem k lepší čitelnosti programu jsem využíval zápisu proměnných ve strukturách, a dále jsem se snažil okomentovávat důležité části programů. Programy jsou založeny jako stavové automaty, ale neodpovídají přesně této definici, protože některé programy se vykonávají paralelně a to odporuje definici stavovým automatům, které mohou být pokaždé pouze v jednom stavu. Je to kompromis mezi přehledností a funkčností programu.

Níže popisuji funkci a účel všech programů. Pro přehlednější popis chodů jednotlivých programů jsem vytvořil zjednodušené obrázky stavových automatů.

### 6.1 Program – Control

Jedná se o hlavní program, který má na starosti nastavení parametrů rotopedu a výběr jednotlivých režimů cvičení. Zde na obrázku je ukázka stavového automatu (obr. 6.1)

Při spuštění se program nachází ve stavu 0 a čeká, dokud nedostane pulz z referenčního indukčního čidla. Děje se tak z toho důvodu, že rotoped po odpojení od napětí ztrácí veškeré informace o poloze klik. A pro další správný chod programu je tato informace nezbytná. V dalším stavu dojde k nastavení všech důležitých parametrů a konstant. Poté plynule přechází do stavu 2. Zde dochází k výběru jednotlivých režimů cvičení. Výběr režimu probíhá pomocí dotykového panelu, kde si můžeme vybrat za pomoci tří tlačítek jeden z režimů. Podle vybraného režimu poté přejde program do stavu 3, 4 nebo 5. Z již zmíněných stavů se lze dostat zpět na výběr režimu (stav 2) třemi způsoby. Stiskem tlačítka **konec režimu** na dotykovém panelu (**konec\_modu**), nesplněním jednoho z požadavků daného režimu (**nesplneni\_modu**) nebo řádného splnění režimu (**splneni\_modu**). Problematikou splnění a nesplnění režimu se ještě budeme zabývat podrobněji dále v kapitole.



Obrázek 6.1: Stavový automat programu Control

## 6.2 Program – Konstantní moment

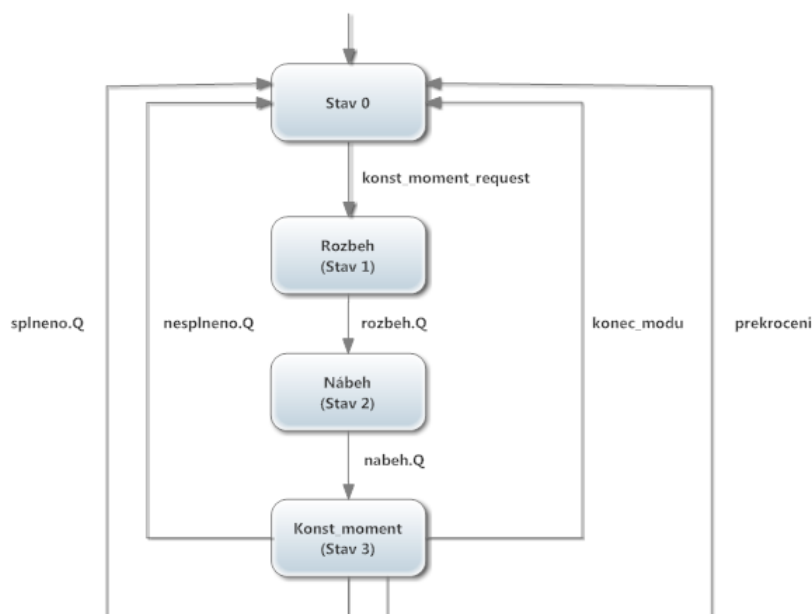
Jedná se o standardní režim každého rotopedu, kdy se nastaví konstantní zátěž, která přetrvává do konce programu. Zátěž lze nastavovat v rozmezí 0 - 55 Nm. Výpočet potřebného proudu vychází z rovnice (1), kterou můžeme pro přehlednost přepsat do rovnice (2), kde  $M$  je námi zadaný brzdňý moment a  $I$  je požadovaný proud do brzdy.

Po spuštění program čeká do doby než se nastaví proměnná *stroj.state.konst\_moment\_request*. Poté přechází do stavu 1. Zde je požadavek, aby se po předem nastavenou dobu kliky otáčely minimální rychlostí 20 ot/min. Důvodem je ověření, zda pacient je schopen šlapat s touto frekvencí, která je potřebná pro správný chod elektrodynamické brzdy. Po splnění přechází do stavu 2, kde probíhá pomalý lineární náběh na předem zvolenou hodnotu brzdňého momentu. V posledním stavu už probíhá normální cvičení. Program se může ukončit čtyřmi způsoby. Uplynutím zvoleného času cvičení. Častým nebo dlouhým poklesem otáček pod minimální hodnotu ot/min. Ukončením pomocí tlačítka **konec programu**.

$$I = 0,0287 \cdot M + 0,3564 \quad (2)$$

## 6.3 Program – Konstantní výkon

V tomto programu na rozdíl od předešlého se udržuje konstantní výkon. To znamená, že na začátku si nastavíme požadovaný výkon a PLC podle aktuálních otáček klik mění proud do brzdy tak, aby výsledný výkon byl neustále konstantní. Brzdňý výkon lze regulovat od 0 do 400 W. Podobně jako v předešlém programu, i zde

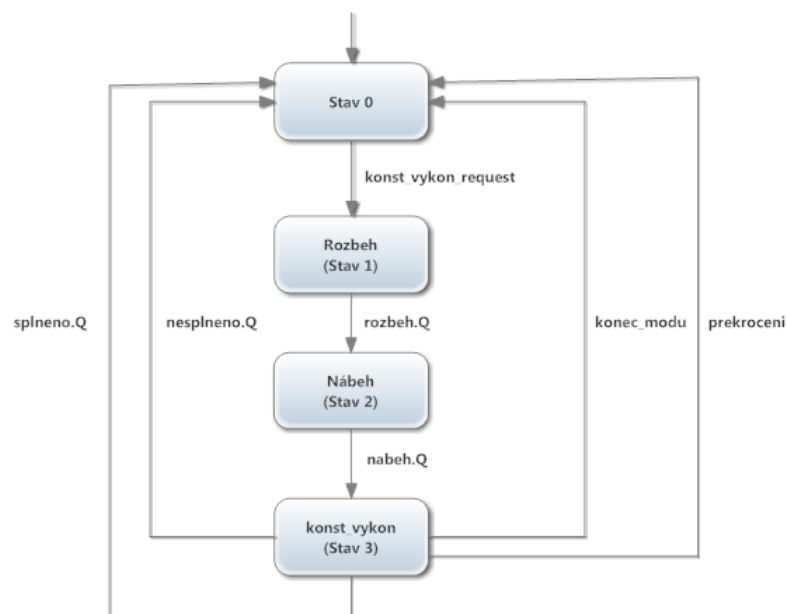


Obrázek 6.2: Stavový automat programu Konstantní moment

se vychází z rovnice (1), kterou upravíme do tvaru (3), kde  $P$  představuje zvolený výkon a  $\omega$  aktuální rychlost klik v  $rad \cdot s^{-1}$ .

Postup programu je téměř stejný jako v předchozím případě jen s malými úpravami. Liší se výpočet proudu do brzdy a minimální potřebné otáčky pro chod tohoto režimu. Minimální otáčky jsou nastaveny na 60 ot/min. Důvodem je to, že při malých otáčkách by se začaly vyskytovat příliš vysoké brzdné momenty a docházelo by k cvičení na maximální zátěž i při malých výkonech. Ukončení probíhá stejně jako u programu Konstantní moment.

$$I = 0,0287 \cdot \left(\frac{P}{\omega}\right) + 0,3564 \quad (3)$$



Obrázek 6.3: Stavový automat programu Konstantní výkon

## 6.4 Program – Elipsa

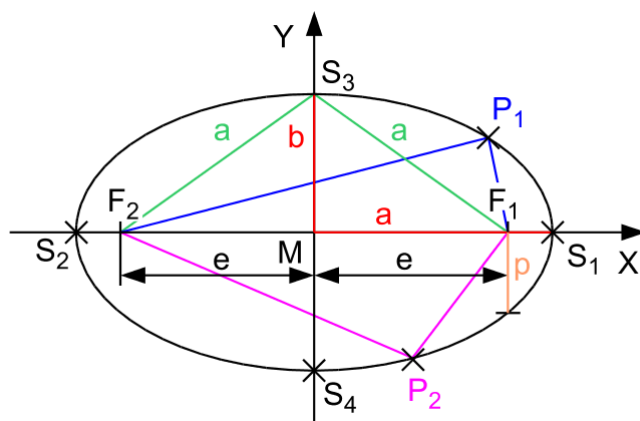
Program Elipsa mění brzdňý moment v závislosti na aktuální poloze klik. Je to v podstatě rozšíření programu konstantního momentu. Výhoda spočívá v tom, že zátěž není všude stejná. A díky tomu se dá nastavit, aby v horní poloze, kde se nachází tzv. „mrtvý bod“ byla zátěž minimální a naopak ve vodorovné poloze byla zátěž maximální.

U elipsy je možné měnit velikost jak hlavní poloosy  $a$ , tak i vedlejší poloosy  $b$ . Pro přehlednost jsou dané parametry k nalezení na obrázku (obr. 6.4). Dále můžeme měnit natočení elipsy. Všechny tyto parametry lze měnit přes dotykový panel více viz. (vizualizace režimu Elipsa).

Pro výpočet zátěže využíváme parametrické rovnice elipsy viz. rovnice (4) a (5), kde  $a$  představuje hlavní poloosu,  $b$  vedlejší poloosu a  $t$  polohu klik ve stupních. Díky těmto rovnicím zjistíme, jakou potřebujeme zátěž na ose  $x$  a  $y$ . Pro získání hodnoty v daném bodě elipsy využijeme rovnici pro zjištění vzdálenosti dvou bodů (6). Zde porovnáваме vzdálenost středu s bodem vypočítaném na elipse. Tato hodnota poté představuje výsledný brzdňý moment, který vložíme do rovnice (2), a tím získáme potřebný proud do brzdy.

Struktura vykonávání a ukončení programu je stejná s předchozími programy. Pozměněn je jenom náběh (stav 2), kde program nejprve vyhodnotí jaká poloosa je menší, a na tu pomalu zvyšuje zátěž. Po dosažení této hodnoty zátěže program čeká, než se kliky opět dostanou do polohy s touto minimální zátěží a plynule přejde do stavu 3, kde už se brzdňý moment mění spojitě v závislosti na poloze klik.

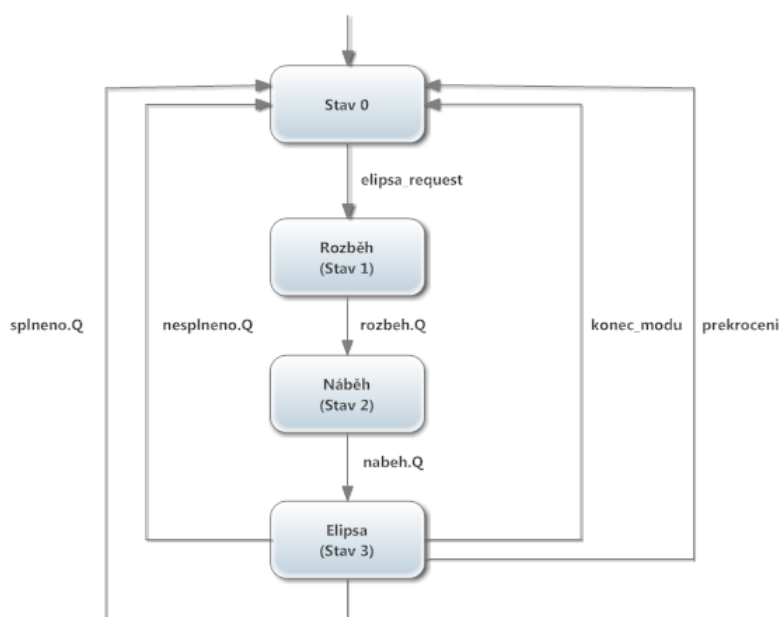
$$x = a \cdot \cos(t) \quad (4)$$



Obrázek 6.4: Popis parametrů elipsy [20])

$$y = b \cdot \sin(t) \quad (5)$$

$$s = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (6)$$



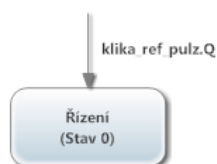
Obrázek 6.5: Stavový automat programu Elipsa

## 6.5 Program – IOControl

Program se spouští po příchodu pulzu z referenčního čidla na klikách. Je to z důvodu, že inkrementální čítač si nepamatuje svoji polohu po zapnutí, a na začátku by mohlo

v některých výpočtech docházet k dělení nulou. Poté se program vykonává neustále až do vypnutí **PLC**.

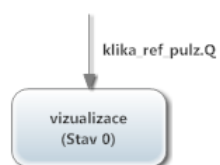
Hlavním úkolem tohoto programu je vypočítávání aktuální rychlosti a polohy klik. Při zjišťování polohy se vyskytl problém v počtu pulzů na otáčku, protože na jednu otáčku inkrementální snímač pošle 33000 pulzů, ale pracuje pouze s typem **integer**. To znamená, že proměnná neustále přetékala. Proto jsem si rozdělil jednu otáčku na dvě části po 16000 pulzech a využíval referenční indukční snímač jako **reset** a kontrolu polohy. Polohu dále přepočítávám na stupně, kde 0° je v horní poloze. Poloha se aktualizuje každých 0,4 ms. Naopak pro výpočet rychlosti otáčení klik je nežádoucí provádět výpočet každých 0,4 ms, proto jsem zde udělal zpoždění, které počítá rychlost jednou za sekundu. Díky tomu se zlepšila přesnost výpočtu i stabilita řízení. Výpočet rychlosti probíhá následovně. Od aktuální polohy klik odečítám minulou hodnotu klik a výslednou hodnotu následně dělím právě 1 s. Program v tomto případě musí ošetřit stavy, kdy za jednu sekundu nastane více jak jedna otáčka, protože by mohlo dojít ke špatnému odečtení poloh. Dále program také vypočítává rychlost otáčení setrvačníku. Posledním část, kterou tento program vykonává je přepočet pomocí lineární interpolace proudu na danou hodnotu v rozmezí 0 až 19660, kde horní mez představuje 3 A, která se následně posílá do motor modulu podle, které se nastaví příslušné hodnoty pro **PWM**. Také je zde ochrana, aby se do brzdy neposlalo více jak 1,77 A.



Obrázek 6.6: stavový automat programu IOControl

## 6.6 Program – HMIControl

Tento program (obr. 6.7) se spouští stejně jako program **IOControl**. Řídí všechny výpočty a funkce potřebné pro vizualizaci. Řeší se zde problematika přepínání jednotlivých stránek vizualizace a režimů. Dále přepočet časů z datového typu **DINT** na **TIME**. Nastavuje signalizační barvu pozadí u prvku **Numeric**, který zobrazuje rychlost otáčení. V případě otáček pod minimální hodnotou nastaví barvu na červenou v ostatních případech na zelenou. Vypisuje informační text na hlavní stránce vizualizace. Uživatel se tak může dozvědět, zda režim splnil nebo ne, popřípadě proč daný režim nesplnil.



Obrázek 6.7: Stavový diagram programu HMIControl

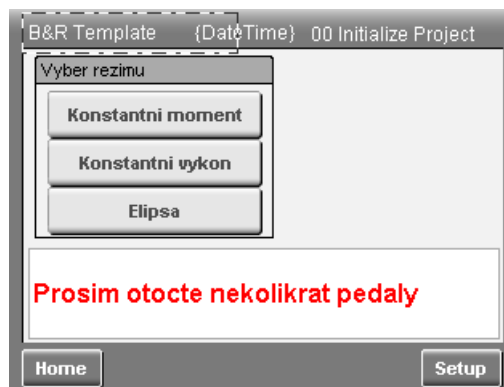
## 7 Vizualizace

Vizualizace se vytváří také ve vývojovém prostředí Automation studio, ale na rozdíl od ostatních funkcí mi nepříjde vizualizace intuitivní a plně funkční v některých případech. Problém nastal, když jsem potřeboval pomocí jednoho ovládacího prvku měnit dvě akce. Původně jsem měl v plánu vybírat jednotlivé režimy pomocí prvku *Dropdown*, ale po propojení s akcemi změny stránky a nastavení proměnné se prvek začal chovat nevyzpytatelně. Někdy došlo pouze ke změně stránky, ale proměnná zůstala nezměněna nebo naopak. Popřípadě došlo k daným změnám až na druhý stisk prvku. K těmto událostem docházelo zcela nahodile. Příčinu se mi nepodařilo zjistit. Proto jsem výběr režimů vytvořil pomocí tlačítek. Z již výše zmíněného problému s propojením více akcí na jedno tlačítko jsem byl donucen pro každý režim přidat ještě tlačítko *start*. Z tohoto důvodu je ovládání trochu neohrabané, ale funkční.

K zobrazování a nastavování hodnot využívám prvek *Numeric*. Pokud se *Numeric* nachází v rámečku nastavení, tak obsahuje funkci *Numpad*, pomocí které je možné měnit jeho hodnotu. Naopak *Numeric* v rámečku informace je určen pouze k zobrazování aktuálních hodnot.

### 7.1 Hlavní strana

Jedná se o hlavní stranu s výběrem jednotlivých režimů. Výběr režimů probíhá pomocí tří tlačítek v horní části. V dolní části vizualizace se nachází prvek *String*, ve kterém se objevují informace o splnění nebo nesplnění daného režimu, popřípadě z jakého důvodu došlo k nesplnění režimu.



Obrázek 7.1: Ukázka vizualizace Hlavní strana



## 7.2 Setup

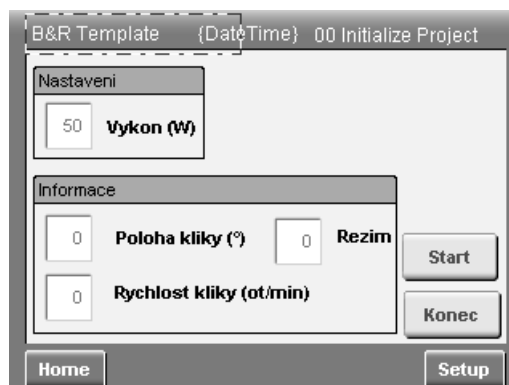
Na této stránce lze měnit časy rozběhu, náběhu, dobu pro splnění režimu a maximální dobu po kterou se může uživatel vyskytovat pod minimálními otáčkami. Dále lze zde nastavit počet poklesu otáček pod minimální hranici. Všechny časy se nastavují v sekundách.



Obrázek 7.2: Ukázka vizualizace Setup

## 7.3 Vizualizace režimu Konstantní výkon

V horní části nastavení lze měnit požadovaný brzdňý výkon. Dolní část je informační. Najdete zde informace ohledně polohy a rychlosti klik. Hodnota režimu odpovídá přímo v jakém stavu se zrovna program nachází viz. (obr. 6.3). *Numeric* zobrazující aktuální otáčky mění barvu svého pozadí v závislosti, zda se otáčky vyskytují pod minimální hodnotou (**červená barva**) nebo nad touto hodnotou (**zelená barva**). Režim se zapíná pomocí tlačítka **Start**. Lze ho ukončit tlačítkem **Konec**. Pro návrat do hlavní nabídky je možné využít tlačítko **Home**.



Obrázek 7.3: Ukázka vizualizace Konstantni výkon

## 7.4 Vizualizace režimu Konstantní moment

Vizualizace režimu Konstantní moment je téměř totožná s vizualizací režimu Konstantní výkon s rozdílem, že zde nastavujeme brzdný moment.



Obrázek 7.4: Ukázka vizualizace Konstantni moment

## 7.5 Vizualizace režimu Elipsa

V horní části můžeme nastavit velikost poloosy **A** a poloosy **B**. Dále lze nastavit natočení elipsy v rozsahu 0 až 360°. Dolní informační část je podobná předešlým dvěma vizualizacím s drobným rozšířením, které nám ještě znázorňuje aktuální brzdný moment na klice.



Obrázek 7.5: Ukázka vizualizace Elipsa

## Závěr

Na rotoped jsem přidal inkrementální snímač a dva indukční senzory pro změření a získání všech důležitých informací. Úspěšně jsem vytvořil program pro řízení rehabilitačního rotopedu, který obsahuje režim konstantního výkonu, režim konstantního momentu a režim eliptického řízení momentu. Dále jsem vytvořil přehlednou vizualizaci pro ovládání rehabilitačního rotopedu přes dotykový panel.

Během této práce jsem absolvoval školení, díky kterému jsem se naučil pracovat s programem Automation studio a psát program ve strukturovaném textu podle normy IEC 61 131. Také jsem si vyzkoušel zapojení celého PLC systému. Díky tomuto tématu diplomové práce jsem si prošel přes všechny fáze práce. Od raných fází výběru vhodného řídicího systému a rotopedu k osazování senzorů a nakonec k vytváření a testování jednotlivých částí programu a vizualizace.

Tato práce skrývá velký potenciál pro další rozšíření. Řídicí program by mohl například ovládat na každé klice jiný brzdný moment. Dále by mohl být k rotopedu přidán motor, který by se používal k tzv. pasivnímu cvičení. Motor by tak pomáhal při rozjezdu jednotlivých režimů nebo při překonávání mrtvých bodů.

## Literatura

- [1] CENDELÍN, Jiří. Historie programovatelných automatů a jejich současné efektivní použití. *Automa* [online]. 2003 [cit. 2014-05-09]. Dostupné z: [http://www.odbornecasopisy.cz/index.php?id\\_document=28831](http://www.odbornecasopisy.cz/index.php?id_document=28831)
- [2] *Eaton elektrotechnika s.r.o.*. 2010 [cit. 2014-05-09]. Dostupné z <http://www.eatonelektrotechnika.cz>
- [3] VONDRA Zdenek. Základy programování PLC. *SPSE Jecna* [online]. [22. 12. 2011] Dostupné z [www.spsejecna.org/skola/documents/Vysledky\\_projektu/PLC\\_zakl.pdf](http://www.spsejecna.org/skola/documents/Vysledky_projektu/PLC_zakl.pdf)
- [4] MARTINÁSKOVÁ Marie, Ladislav Šmejkal. *Řízení programovatelnými automaty*. Praha : Vydavatelství ČVUT, 1998.
- [5] KOVÁŘ Josef, Zuzana Prokopová, Ladislav Šmejkal. Programování dle normy IEC 61 131. *SPS Zlín* [online]. [9. 5. 2014]. Dostupné z [http://www.spszl.cz/soubory/plc/programovani\\_dle\\_normy\\_iec61131.pdf](http://www.spszl.cz/soubory/plc/programovani_dle_normy_iec61131.pdf)
- [6] MARTINÁSKOVÁ Marie. Programovací jazyky pro PLC. *Automa* [online]. [28. 12. 2011]. Dostupné z [http://www.odbornecasopisy.cz/index.php?id\\_document=28831](http://www.odbornecasopisy.cz/index.php?id_document=28831)
- [7] Ladder diagram. *Automation & Drive* [online]. 2009 [cit. 2014-05-13]. Dostupné z: <http://www.automation-drive.com/ladder-diagram>
- [8] Good Function Blocks, Bad Function Blocks. *Contact and Coil*. 2012 [online]. [29. 4. 2012]. Dostupné z <http://www.contactandcoil.com/tag/function-block/>
- [9] Structured Text. *Messung* [online]. 2008 [29. 4. 2012]. Dostupné z [http://www.messung.com/PLC\\_14\\_nexgen2000plus\\_programming.asp](http://www.messung.com/PLC_14_nexgen2000plus_programming.asp)
- [10] FRUMENTO, Ch., MESSIER, E., MONTERO, V.. History and future of rehabilitation robotics. *Worcester Polytechnic Institute*, 2010. Dostupné z: [http://www.wpi.edu/Pubs/E-project/Available/E-project-031010-112312/unrestricted/HRRIQP\\_Final.pdf](http://www.wpi.edu/Pubs/E-project/Available/E-project-031010-112312/unrestricted/HRRIQP_Final.pdf)
- [11] HILLMAN, Michael. Rehabilitation robotics from past to present - a historical perspective. *Bath Institute of Medical Engineering*, 2003. Dostupné z: < [http:](http://)

- [//www.researchgate.net/publication/225320978\\_2\\_Rehabilitation\\_Robotics\\_from\\_Past\\_to\\_Present\\_\\_A\\_Historical\\_Perspective](http://www.researchgate.net/publication/225320978_2_Rehabilitation_Robotics_from_Past_to_Present__A_Historical_Perspective)
- [12] PLC Twido Training with Schneider Electric Training Centre. *Galih Eka Priminta* [online]. 2012 [cit. 5. 2014]. Dostupné z: <http://galihekapriminta.blogspot.cz/2012/04/plc-twido-training-with-schneider.html>
  - [13] Léčba s přístroji MOTREN - ROTREN. *Kalpe* [online]. 2014 [cit. 2014-04-18]. Dostupné z: <http://www.kalpe.cz/lecba-s-pristroji-motren-rotren/>
  - [14] Whirlwind Fan Air Resistance Exercise Bike. *HSN* [online]. 2014 [cit. 2014-05-15]. Dostupné z: <http://www.hsn.com/products/proform-whirlwind-fan-air-resistance-exercise-bike/6933736>
  - [15] What are the different types of resistance on exercise bikes?. *Reviews.com* [online]. 2014 [cit. 2014-04-17]. Dostupné z: <http://www.reviews.com/exercise-bikes/what-are-the-different-types-of-resistance-on-exercise-bikes/>
  - [16] Cyklotrenažéry. *JAFa Squash* [online]. 2014 [cit. 2014-04-17]. Dostupné z: <http://www.rotopedy-fitness.cz/clanky/cyklotrenazery>
  - [17] *B&R* [online]. 2014 [cit. 2014-04-25]. Dostupné z: <http://www.br-automation.com/cs/perfection-in-automation/>
  - [18] BES M08MI-PSC15B-BV02. In: *Balluff* [online]. 2013 [cit. 2014-05-15]. Dostupné z: [http://www.balluff.com/media/DE\\_EN\\_148852.pdf?lang=en](http://www.balluff.com/media/DE_EN_148852.pdf?lang=en)
  - [19] Inkrementální rotační snímače IRC 300 až 325. In: *Larm* [online]. 2014 [cit. 2014-05-15]. Dostupné z: [http://www.larm.cz/pages/larm\\_html\\_cz/pdf/irc300-325.pdf](http://www.larm.cz/pages/larm_html_cz/pdf/irc300-325.pdf)
  - [20] Elipsa. *Wikipedia* [online]. 2014 [cit. 2014-05-15]. Dostupné z: <http://cs.wikipedia.org/wiki/Elipsa>

## Seznam obrázků

1.1	Kompaktní PLC [2]	12
1.2	Modulární PLC [2]	12
1.3	Činnost PLC	15
1.4	Ukázka příčkového diagramu [7]	17
1.5	Ukázka funkčního blokového schématu [8]	18
1.6	Ukázka seznamu instrukcí [12]	18
1.7	Ukázka strukturovaného textu [9]	19
2.1	Příklad přenosného rehabilitačního zařízení [13]	21
2.2	Příklad rehabilitačního zařízení poháněný silou asistenta [13]	22
3.1	Příklad čelistové brzdy [16]	23
3.2	Příklad elektromagnetické brzdy	24
3.3	Příklad brzdy pomocí odporu vzduchu [14]	25
4.1	Ukázka řídicího PLC [17]	27
5.1	Ukázka závislosti výkonu na proudu	30
5.2	Ukázka závislosti momentu na proudu	31
5.3	Princip PWM modulace	31
6.1	Stavový automat programu Control	33
6.2	Stavový automat programu Konstantní moment	34
6.3	Stavový automat programu Konstantní výkon	35
6.4	Popis parametrů elipsy [20])	36
6.5	Stavový automat programu Elipsa	36
6.6	stavový automat programu IOControl	37
6.7	Stavový diagram programu HMIControl	38
7.1	Ukázka vizualizace Hlavní strana	39
7.2	Ukázka vizualizace Setup	40
7.3	Ukázka vizualizace Konstantni výkon	40
7.4	Ukázka vizualizace Konstantni moment	41
7.5	Ukázka vizualizace Elipsa	41

## Příloha

Příloha obsahuje CD na zadních deskách. Součástí CD je:

- Text diplomové práce ve formátu **.pdf**.
- Zdrojový kód programu.

